A Secure Online Key Establishment Scheme for Mobile Heterogeneous Sensor Networks

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 925479, 12 pages
http://dx.doi.org/10.1155/2014/925479

Research Article

A Secure Online Key Establishment Scheme for Mobile Heterogeneous Sensor Networks

Sarmad Ullah Khan,[1] Claudio Pastrone,[2] Luciano Lavagno,[3] and Maurizio A. Spirito[2]

[1] *Electrical and Telecommunication Department, CECOS University, Peshawar, Pakistan*
[2] *Pervasive Technologies Research Area, Istituto Superiore Mario Boella (ISMB), Turin, Italy*
[3] *Electronics and Telecommunication Department (DET), Politecnico di Torino, 10129 Turin, Italy*

Correspondence should be addressed to Sarmad Ullah Khan; sarmadullahkhan1@gmail.com

Received 15 May 2014; Revised 16 September 2014; Accepted 9 October 2014; Published 4 November 2014

Academic Editor: Jiun-Long Huang

Advancement in wireless sensor network (WSN) technology makes it more attractive technology to be adopted in real world applications including home and industrial automation and e-health or infrastructure monitoring. However the mobility consideration in these applications makes the security requirements an essential and challenging task. To make the communication secure and the network robust against attacks, a proper key management scheme should be used. This paper presents a secure online key establishment and authentication scheme for mobility supported heterogeneous sensor networks. The performance evaluation of the proposed scheme is done using OMNET++ simulator while AVISPA tool is used to validate the security of the proposed scheme. It is clear from the obtained results that the proposed scheme provides better network connectivity at a low cost of memory occupation and communication overhead during the authentication and key establishment phases while improving its network resilience against attacks.

1. Introduction

Wireless sensor networks (WSNs) contain a large number of small and resource constrained devices. They have limited computational power, communication range, and memory/storage capabilities. However their low cost and easily adopted nature in various environments make them suitable to be used in home and industrial automation, environmental monitoring, and e-health and military applications. The deployment of WSNs can be in a controlled environment (e.g., home/industry) or in uncontrolled areas (e.g., public spaces/hilly areas).

Since WSNs have resource limitation characteristics in terms of available bandwidth and communication range and have self-organizing and reconfiguration capabilities, they may introduce many challenges with respect to secure communication. In order to achieve a certain level of security, proper security measures must be taken to protect the network against possible attacks. Resource limited nature of WSNs becomes an important hurdle in the adoption of traditional security approaches. Therefore, a completely new mechanism to secure WSNs needs to be defined which suits its constrained nature. As cryptography plays an important role in secure communication, a proper key management should be considered.

A number of key management schemes have been proposed in the literature. Initially asymmetric cryptography and elliptic curve cryptography (ECC) were considered an expensive solution to the sensor applications due to their high computational cost, energy consumption, and storage requirements. However, research in [1, 2] showed that ECC (having low computational cost and small key size) and Rabin's scheme (supporting fast encryption/decryption time compared to RSA) can be used for sensor networks.

Wireless sensor networks can be divided into homogeneous sensor networks and heterogeneous sensor networks. Homogeneous sensor networks consist of nodes having the same capabilities and have been the first choice for the researchers for developing the security algorithms. But the studies of [3, 4] showed that these networks have scalability

issues and performance limitations which have been proved by both theoretical [5] and simulation analyses [4]. These limitations open the door for developing heterogeneous sensor networks (HSNs) consisting of a number of different nodes having different varying capabilities [6]. For instance, energy and link heterogeneity in HSNs have been proposed to increase throughput and network life time without increasing the cost by Yarvis et al. [7] while the energy consumption and network life time have been analyzed by proposing a periodic data transmission from sensing field to remote receiver in [8].

In this paper, we propose a secure online authentication and key establishment and management approach for HSNs. The objective of this paper is not only to increase the network life by reducing the energy consumption with the help of optimized message exchanged pattern during the authentication phase and key establishment phase but also to provide better network connectivity and resilience against attacks compared to the existing approaches. The paper is organized as follows. Section 2 presents a brief literature review while the proposed scheme is discussed in Section 3. Section 4 presents the security validation of the proposed scheme using AVISPA tool against some well-known attacks while the performance evaluation using OMNET++ simulator is performed in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

Here we present a brief overview of some of the existing well-known key management schemes proposed for the heterogeneous WSNs and homogeneous WSNs. Basically the key management schemes have been divided into two broad categories: (1) centralized approach and (2) distributed approach. In the centralized approach, a more powerful node (base station) is responsible for holding and managing the secret keys. Public key cryptography is a well-known example of the centralized approach. However the public key approach does not suit the resource constrained WSNs. However to adopt the centralized approach, a centralized keying scheme has been presented by Perrig et al. [9] called SPINS, where each node is assigned a secret key while its corresponding key is kept at the network controlling entity. However the key release mechanism consists of one-way hash chain and epoch delay for the authentication purpose in broadcast scenario. Hence there is no common key between any two nodes and base station or network controlling entity plays an important role in establishing a key between the two communicating nodes. For instance, node x and node y want to communicate with each other. Node x sends a request to node y which will be forwarded to base station by node y. The base station generates a secret key for x and y. The base station encrypts it with the secret keys that it shares with x and y. Since base station acts as trusted server to establish a secret key, the scheme will fail if the base station fails.

In the distributed approach, each node is assigned secret key(s) before the network deployment and the nodes use those keys for secure communication with each other. Symmetric key approach is one of the best examples of distributed key approaches. However this approach occupies lot of memory especially in large network scenario. To reduce this memory cost, a random key predistribution scheme has been proposed in [10] which does not require trusted authority for establishing a key between any two nodes of the network. In this scheme, each node is assigned a set of randomly selected keys called a key pool from a large pool before the network deployment. For secure communication, two nodes directly share their assigned keys identity with each other to find a common secret communication key. However to make this scheme more secure, Chan et al. [11] introduced the "q-keys" concept. According to this scheme, the two communicating nodes must share at least q-keys for establishing a secret communication key. But this scheme has a large memory cost in storing a large number of keys in each node. To improve the network performance and connectivity, Liu et al. [1] presented a key establishment scheme based on the deployment knowledge of the nodes in a network coupled with Rabin's scheme [12] to make it resilient against attacks. A pairwise key predistribution scheme called NPKPS is presented by Zhang et al. [13] to achieve better security, network connectivity, and less memory cost.

For heterogeneous sensor networks, an unbalanced key management scheme was proposed by Du et al. [14] to increase network connectivity, reduce memory cost, and increase the network resilience compared with balanced key predistribution approaches. In this scheme, network consists of high capability nodes assigned m keys and low capability nodes assigned l keys such that $m \gg l$. To further reduce the memory cost while keeping the security level constant, two-key pool approach was proposed by Khan et al. [15] for secret key generation. A combinatorial design for key distribution is presented by Çamtepe and Yener [16] while the finite projective plane (FPP) design is presented by Sánchez and Baldus [17] to distribute polynomial shares. This approach makes it possible to establish a direct pairwise key for a large number of nodes without considering their physical connectivity in the network. To support mobility in WSNs, Maerien et al. [18] presented the management of secret keys protocol in which each node is assigned only one symmetric key shared with its server.

A group-based key management scheme was presented by J. Zhang and L. Zhang [19] for heterogeneous sensor networks in which a large number of keys are divided into small groups of keys while each small key-group is assigned to each group of nodes or cluster while Du et al. [2] presented an elliptic curve-based routing driven key management scheme for sensor networks. Results show that elliptic curve approach provides better network resilience against attacks and better network connectivity than the key predistribution approaches. However key predistribution or symmetric key distribution based on PKI approach provides low memory cost and better resilience against attacks [20] if their deployment location is known in advance. To support node mobility in key predistribution approaches, efficient key management schemes were proposed in [21, 22], which not only increase the security level but also reduce the memory cost and computational overhead. Kyeong and Ramakrishna [23] proposed a level-based key management scheme for

mobility supported sensor networks for multicast communication while Chuang et al. [24] proposed two-layered dynamic key management scheme for mobility supported networks and Blundo et al. [25] presented a polynomial-based key predistribution scheme.

The online key generation approach is good for large WSNs. This saves the memory cost of each node, but it consumes some energy in generating the secret key. Since the keys are not stored in each node, it has high network resilience against node compromise attacks as compared to previously described approaches. One of the online key generation approaches for cluster-based sensor network is presented in [26] which provides better network resilience and has low memory cost while another online key generation approach is presented in [27] where each node is assigned small number of generation keys for online key generation. In this approach, two communicating nodes contact their network manager or cluster head to discover a shared generation key between them for secret key generation. Node addition and revocation capability were introduced by Poornima and Amberker [28] in a tree-based key management scheme while unpredictable software-based attestation solution (USAS) is presented in [29] to detect the compromised node in the network.

Zhang et al. in [30] proposed random perturbation-based scheme for pairwise key establishment in sensor networks in which nodes establish secret key with each other without revealing their secret to each other. The basis of this scheme is [25]. However noninteractive pairwise key establishment was proposed for sensor networks that provides high resilience and network connectivity.

3. Proposed Scheme

The focus of the proposed scheme is to generate mutual authentication key and mutual secret key in runtime. The network architecture of the proposed scheme is based on heterogeneous sensor networks. However three different types of nodes are selected for the proposed scheme: (1) base station (BS), (2) fixed nodes (FNs), and (3) mobile nodes (MNs). The capabilities of these nodes differ in terms of computational power, memory space, and energy resources while the communication of all nodes is kept constant to ensure one-hop communication links. The base station and the fixed nodes are made powerful nodes to manage the networks and are also provided with tamper-resistant hardware, while the mobile nodes are considered less powerful nodes whose function is to gather the information and forward it to the BS through FNs. Here the mobility is introduced by making the MNs mobile that change their positions in the network according to the specific mobility model (mentioned in Section 5). Each type of nodes has different functionality and is given different task in generation of secret keys. The functional block diagram of each type of node is shown in Figure 1.

It is clear from the functional block that base station communicates with only fixed nodes while mobile nodes communicate only with the fixed nodes as well. Hence the base station acts as a trusted server for the fixed nodes while the fixed nodes act as trusted server for the mobile nodes.

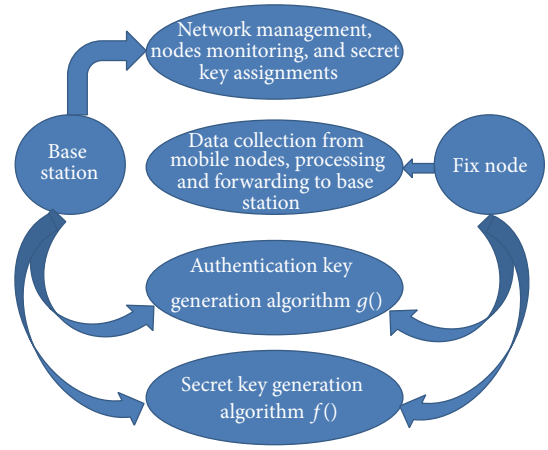

Figure 1: Functional block diagram of base station and fixed node.

In addition, cluster-based network architecture is adopted in which each fixed node acts as a cluster head. In the next subsections, we describe the proposed key generation algorithm, key predistribution, and proposed cluster formation approach followed by authentication and key establishment phases.

3.1. Key Generation Algorithm. Basically we are using two different types of keys: (1) authentication key and (2) secret communication key. Using some input parameters and key generation algorithms generates these two keys. The BS has the knowledge of all those input parameters and generation algorithms. The authentication key (K_{auth}) is generated using

$$K_{\text{auth}} = g\,(\text{FNPNS; random number; SKG})\,, \tag{1}$$

where $g(\cdot)$ is an authentication key generation function, FNPNS is fixed nodes prime number sum, and SKG is a secret key generation code while the secret key (SK) is generated using

$$\text{Secret key} = f\left(\text{PN}_1; \text{PN}_2; \text{MNPN; SKG}\right), \tag{2}$$

where $f(\cdot)$ is a secret key generation function, MNPN is a mobile node prime number, PN_1 is a generated prime number-1, and PN_2 is a generated prime number-2. BS distributes the generated authentication key and the secret key among the deployed nodes of the network to secure network against attacks. The distribution of these authentication keys and secret keys are discussed in the next subsection.

3.2. Key Predistribution. After the generation of the authentication key and the secret key, BS assigns the authentication key (K_{auth}) and the secret key (SK) to each MN along with the network public key K_{plc} before the network deployment. Each MN is also assigned sum of the prime numbers of fixed nodes (FNPNS), a prime number (MNPN), and a random number used in generating the authentication key.

In order to make the fixed nodes a trusted server for the mobile nodes and to enable their secure communication with both base station and mobile nodes, they are provided

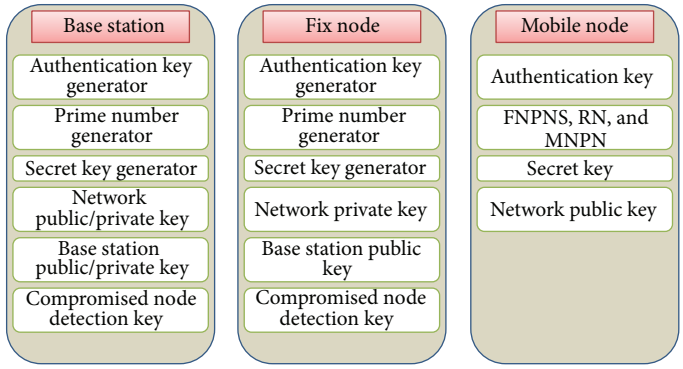

Figure 2: Predistribution of key materials to each node.

with the public/private key pair, base station's public key, authentication key generation function $g(\cdot)$, compromised node detection key (CNDK), secret key generator (SKG), secret key generation function $f(\cdot)$, and network private key (K_{prt}) along with a prime number. Figure 2 shows the predistribution of key materials to each node.

3.3. Cluster Formation. Once the network is deployed, each fixed node broadcasts Hello messages to know about its member mobile nodes. However the broadcast attempts are limited in the proposed scheme (e.g., 3 times during simulation in the proposed scheme) in order to advertise their presence to neighboring MNs. The FNs discovery is thus passively performed by the MNs: this approach has been selected to limit the messages exchange overhead since it is expected that the number of FNs is lower than the number of MNs. Such Hello messages have node IDs and a nonce (used for authentication purpose) encrypted by k_{prt}. To make the network connected, the FNs are deployed in such a way that each mobile node receives Hello messages from more than one fixed node. The MN can then select a given FN as relevant CH depending on Hello message signal strength. This helps each MN to create a list of its neighboring fixed nodes. A mobile node in handover scenario uses this list. In case the mobile node does not receive any fixed node Hello message within the specified interval of time, it starts broadcasting its own Hello messages to discover its neighboring fixed nodes. These messages include a nonce encrypted by k_{plc} (see Section 4.1 for an explanation of why this approach counteracts denial of service attacks). Figure 3 describes the assumed virtual network organization.

3.4. Authentication Phase. Both the CHs and the MNs need to authenticate each other during the network initialization phase. The network public/private key pair is used to authenticate that the CH authentication key assigned to each MN is used in the authentication of each MN with the help of key generation algorithm assigned to each CH. Furthermore, mobile node encrypts the joining request using the network public key and sends it to the CH. This joining request includes a random number, the FNPNS, received nonce from CH during cluster formation phase, and encrypted prime number using the authentication key. After receiving this

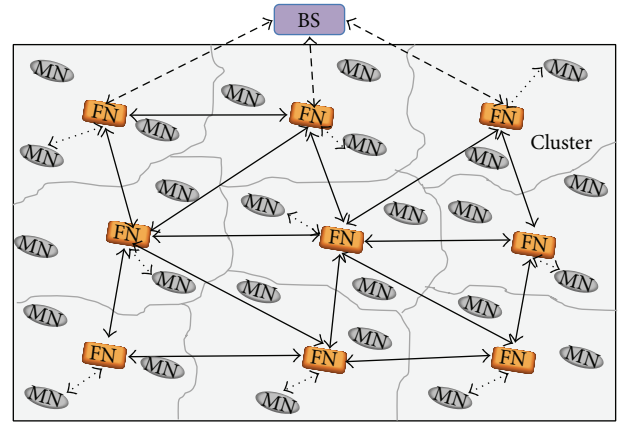

Figure 3: Virtual network topology.

information, CH generates the authentication key of the MN using key generation function $g(\cdot)$.

Once the CH successfully generates the authentication key, it decrypts the received encrypted nonce to verify the authenticity of the MN. Successful decryption provides the CH with the MN's prime number. The obtained prime number will be used in generating the secret key for that particular MN. As a next step, the CH generates the network authentication code (NAC) for the successfully authenticated MN and sends it back with the joining confirmation message. The NAC is a combination of BS prime number and FNs prime number with the addition of the MN ID (i.e., NAC = h(BS Prime Number, FNs Prime Number, MN ID), where $h(\cdot)$ is a one-way hashing function). This approach reduces the authentication overhead during the handover phase of a MN from one cluster to other clusters within the same network. However to protect the NAC from unauthorized access or attacks, it is periodically updated. As mentioned previously, each CH is assigned a fast key revocation algorithm [31] to protect the assigned materials for authentication key generation and secret key generation.

3.5. Key Establishment and Management. Before the network deployment and initialization phase, each MN is assigned a secret key for secure communication with its CH while the

CH is provided only with the key generation algorithm. The CH received the required information for secret key generation from the MN during the authentication phase (e.g., MN prime number). The CH uses the received information and secret key generator to generate the first component as first prime number (PN_1) of the secret key. This generated PN_1 is used to generate a second prime number (PN_2) with the help of received prime number and SKG. After the generation of PN_1 and PN_2, the CH generates the secret key of the joining MN for secure communication using key generation function $f(\cdot)$. This generated secret key will be the same secret key assigned to the MN before the network deployment.

Since the MNs are movable and need to communicate with each other as well, there must be a secret key for secure communication among the MNs. This is achieved by generating a secret key for any pair of MNs through their CH. Here it is assumed that the two MNs belong to the same cluster. For instance, a secret key for secure communication between the mobile node A and the mobile node B is generated using their identities ID_A and ID_B, the prime numbers of each MN received by the CH during the authentication phase, and secret key generation function $f(\cdot)$. Once the key is generated, CH encrypts it with the key that it uses for secure communication with the MN and sends it to both the MNs separately.

3.6. Mobile Node Handover. The network is deployed such that each MN receives Hello messages from more than one FN during the cluster formation phase. The selection of CH is based on its signal strength. During the movement of a MN, each MN periodically checks the signal strength of its cluster head which is not compulsory in case of static networks. This helps the MN to remain connected in the network. Once the MN detects that its signal strength drops below a predefined threshold value, it starts searching for new CH by broadcasting its own Hello messages. In response to those Hello messages, the MN will update its neighboring FNs list and will select the FN with the highest signal strength as its new CH at that particular instant. After the selection of new CH, the MN sends a leaving message to its old CH and a joining request to the new CH. The leaving message includes the new CH ID while the joining message includes the NAC for the authentication purpose. Once the NAC is authenticated, the new CH will get the joining MN's prime number from its previous CH using the secure link established among the CHs during the network initialization phase. After receiving the MN's prime number, the new CH generates the secret key of that MN. Both CHs then update their member MNs table and send the updated table to the BS to avoid node replication attacks.

4. Security Analysis

Since cryptography is considered as the main building block of any security primitive, the cryptographic keys should also be secured and authentic. To this aim, the key management scheme should be secure and each node of the network should be able to authenticate the cryptographic key(s). This is the most challenging problem in the considered resource constrained networks. In order to validate the secrecy of the proposed key management scheme for heterogeneous sensor networks, we used the AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [32]. AVISPA is a push-button tool for the automated validation of Internet security-sensitive protocols and applications. It provides a modular and expressive formal language for specifying protocols and their security properties and integrates different back ends that implement a variety of state-of-the-art automatic analysis techniques (e.g., OFMC, ATSE, etc.). We implemented the proposed key management scheme in AVISPA and checked its security using some of the attacks provided by AVISPA, namely, OFMC (on-the-fly model checker) and CL-AtSe (constraint-logic-based attack searcher). The former builds the infinite tree defined by the protocol analysis problem in a demand-driven way, that is, on-the-fly, and uses a number of symbolic techniques to represent the state space. The latter provides a translation from any security protocol specification written as a transition relation into a set of constraints which can be effectively used to find attacks on protocols. Both translation and checking are fully automatic and internally performed by CL-AtSe; that is, no external tool is used. In this approach, each protocol step is modeled by constraints on the adversary knowledge. These results are shown in Table 1.

Table 1: AVISPA simulation results.

Technique	Summary
OFMC	Safe
CL-AtSe	Safe

We evaluated the behavior of the proposed scheme when exposed to some well-known attacks including man-in-the-middle attacks (used to launch replay attacks, wormhole attacks, and denial of service attacks), node replication attacks, and Sybil attacks.

4.1. Man-in-the-Middle Attacks. The man-in-the-middle attack is common and easy to be implemented in WSNs where the deployment is not secure and can be accessed by an attacker easily. This attack can be used to launch other attacks such as replay attacks (denial of service (DoS) attacks), wormhole attacks, and black hole attacks. First we describe the effect of replay attacks used to launch DoS attacks followed by wormhole attacks.

4.1.1. Denial of Service Attacks. The denial of service (DoS) attacks are used to degrade the performance of a network by exhausting its resources, such as bandwidth, memory, or processor time, for example, by sending fake network topology and routing information. DoS attacks in the proposed scheme might be performed during (1) the cluster formation phase, (2) the MN transition phase from one cluster to another, and (3) the addition of new nodes in the network.

During the cluster formation phase, all FNs periodically broadcast Hello messages for a specific number of times (3 times in the proposed scheme) and each Hello message is encrypted by the network private key. If an intruder

Table 2: DoS attack evaluation.

Total number of attacking nodes	Total Hello messages sent by the attacker	Total received Hello messages from attacker
1	4844	16
2	9267	67
3	14044	174

broadcasts its own Hello messages, those messages will not be decrypted by the network public key and would be discarded by the MNs of the network. Also the MNs check how many Hello messages they received from a specific FN. If those messages are above a predefined threshold, the MNs consider FN as an adversary or malicious node of the network. We implemented a DoS attack by replaying the FNs Hello messages in the OMNET++ simulator and we have analysed the performance of the proposed scheme by introducing a different number of attacking nodes. In the evaluated DoS scenario, the attacker captures the Hello messages of the FNs and forwards it to the MNs by changing the source node ID of the packet. We have analyzed how many modified Hello messages are received by MNs from the attacker. Table 2 shows the results of the simulation with 200 MNs and 16 FNs and different number of attacking nodes.

In the resulting simulation scenarios, we change the attacking node ID during the simulation. This is important because of random positioning of nodes in the networks and their effects on the neighboring nodes. Hence each node of the network acts as an attacker for short time. When the attacking node changes its ID, the FNs again start broadcasting Hello messages towards their neighboring MNs. The changing of ID is sequential and for a specific time (in the proposed scenario, it is 10 seconds).

During the MNs transition from one cluster to another cluster, each MN broadcasts Hello messages to know about its neighboring FNs. Since, in the proposed scheme, these Hello messages also include the NAC (network authentication code) which is used by the FNs of the network to authenticate the incoming MN, the adversary would not be able to send the correct NAC to the FNs and would be detected at its first broadcast. The adversary can also add a fake node in the network to try to get access to the network by broadcasting a Hello message to know about its neighboring FNs. But since in the proposed scheme the BS informs the FNs about the addition of a new MN, its ID, and a specific authentication code assigned to that MN, the adversary fake node would not be able to authenticate itself to the FNs and would be detected at its first broadcast. Thus the proposed scheme effectively avoids these three different types of DoS attacks that could be launched by an adversary at any stage of the network and could exhaust the resources of both the FNs and the MNs.

4.1.2. Wormhole Attack. In the wormhole attack, an adversary launches two nodes in two different clusters, connecting them using a direct communication link called wormhole link. This link could be an Ethernet cable, long-range wireless transmission, or an optical link. The main purpose of this attack is to capture the traffic of one part of the network and replay it in the other parts of the network. Also this attack is easily implemented in multihop networks. This attack can be launched against the proposed scheme during the initialization phase by replaying the Hello messages of one part of the FNs of the network into another part of the network to attract the MN communications. However in this type of attack, the attacker acts as a man-in-the-middle and just forwards the packets from one part of the network to the other but would not be able to understand or extract the key/data information from the received packets. The verification of this man-in-the-middle attack was performed using the AVISPA tool which verified the security of the proposed scheme as shown in Table 1. Note that we use a single hop network topology approach in which each MN is only one hop away from the FNs. The FNs send their member MN lists to the BS and the BS knows the location and position of each cluster of the network, so this wormhole attack, if launched after the network initialization phase, can be easily detected and avoided in the network by the BS. The adversary node in one cluster cannot pretend to be a member node of another cluster (even of neighboring cluster) despite having updated information received by the BS from the FNs.

4.2. Node Replication Attacks. The MNs are more vulnerable than the FNs and can be easily captured, analyzed, and replicated by the attacker in various positions of the network. Such attacks may allow the adversary to corrupt data and may disconnect a significant part of the network. Node replication attack might be possible (1) during the network initialization phase and (2) after the network initialization phase. However node replication in the network initialization phase is difficult for an attacker because of the secure deployment phase. The attacker can launch the node replication attack after the network initialization phase when the network will no longer be under observation by the network deployer. Since in the proposed scheme the MNs communicate directly with their selected FNs/CHs, each FN sends its MN member IDs to the BS after the initialization phase. Also, during the transition phase of a MN from one cluster to another cluster, the new CH verifies the transition of the incoming MN from its previous CH. Thus the BS immediately detects node replication in the network during the initialization phase or by the FNs during the handover phase of the MNs. Thus the proposed scheme avoids node replication attacks in the network.

4.3. Sybil Attack. In the Sybil attack, a malicious/attacker node assumes multiple identities to launch attacks against storage space of its neighboring node or some protocol specific attacks (e.g., routing algorithms). This attack is reactively successful against key predistribution mechanisms, but since the proposed scheme uses an online authentication key and secret key generation technique that involves the node ID and its unique prime number, it makes it difficult for an attacker to launch Sybil attack. For example, in other key predistribution approaches, if an attacker compromises a few nodes and obtains a few authentic keys of the network, it can launch Sybil nodes with different IDs and can assign

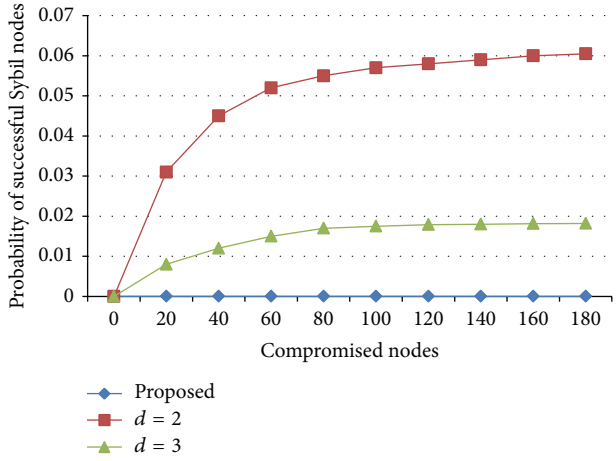

Figure 4: Probability of successfully generated Sybil nodes.

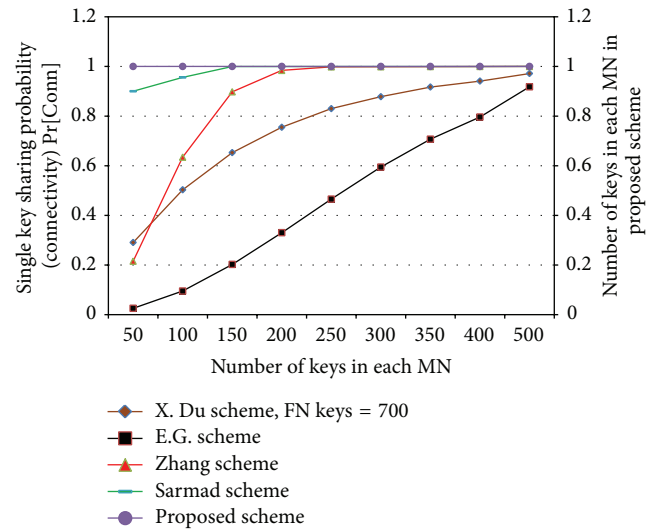

Figure 5: Probability of sharing at least one common key (connectivity).

them those compromised keys. Now when the verification and authentication process starts for some authentic nodes of the network, the Sybil nodes can give them a proof of authenticity by sending their own key pool IDs along with the compromised key IDs. If the verifier node and the Sybil node have a common key among their key pool (i.e., the key ID and the actual key are the same), the verifier node would not be able to detect the Sybil node. Otherwise if the key IDs match but the keys do not match, then the verifier node can detect such Sybil nodes in the verification process. But in the proposed scheme there is no concept of initial secret key predistribution and all the predistributed keys are the function of the node IDs and their assigned secret prime number. Hence node compromise does not help the attacker to launch Sybil nodes with fake IDs.

In the key predistribution approach [33], if every node is assigned "k" keys from a key pool of size "m" and "d" verifiers are used to verify a node, and if an attacker compromises "c" nodes to create a compromised key pool of size "n," then the probability of a Sybil node to be successful is

$$\text{Probability} = \sum_{t=1}^{k} \frac{\binom{n}{t}\binom{m-n}{k-t}}{\binom{m}{k}} \left(\frac{\binom{m-k+t}{k}}{\binom{m}{k}} \right)^{d}. \quad (3)$$

Figure 4 shows the probability of successfully generated Sybil nodes in the proposed scheme compared with scheme [33].

5. Performance Evaluation

The performance of the proposed scheme is analyzed using the OMNET++ simulator, in terms of network connectivity, network resilience against node capture attacks, energy consumption, memory cost, and communication overhead. Its security validation is done using the AVISPA tool. The simulation results have been obtained using OMNET++ 4.1 with the mobility framework MiXiM 2.0.1. A network composed of 500 MNs and 16 FNs defines the simulation scenario. The size of the network simulation area is 400 m × 400 m. Both the FNs and the MNs use the 802.15.4 CSMA and radio specification based on the CC2420 radio chip. The transmission power is set to 10 mW and the sensitivity is set to −95 dBm for all nodes. The random walk mobility model in which the speed of the MNs is constant describes the mobility of the MNs, but a random direction (within a predefined range) is chosen periodically. More specifically, the speed of the MNs is set to 1 m/s (i.e., human walking speed) and their direction update interval to 0.1 s. The simulations were repeated 3 times for 5000 seconds for each considered scenario.

5.1. Network Connectivity. In order to show the effectiveness of the proposed solution in terms of network connectivity, the simulation results of the proposed scheme are compared with [10, 14, 15, 19] where the connectivity depends on the key sharing probability. For a balanced key predistribution scheme, the single key sharing probability between the MN and the FN is given by

$$\Pr[\text{Conn}] = 1 - \frac{(P-K)!\,(P-K)!}{P!\,(P-2K)!}, \quad (4)$$

where K is the number of keys assigned to FNs and MNs from a pool of P keys. Instead, for the unbalanced key predistribution schemes [14, 15], the single key sharing probability is given by

$$\Pr[\text{Conn}] = 1 - \frac{(P-K)!\,(P-S)!}{P!\,(P-S-K)!}, \quad (5)$$

where K is the size of the key pool assigned to each MN and S ($S \gg K$) is the size of the key pool assigned to each FN. In the proposed scheme, the K_{plc} is assigned to each MN and the K_{prt} to each FN before network deployment. These two keys allow connecting a MN to an authentic FN of the network. Hence the connectivity of the network is almost 100% if and only if the FN is not compromised. Figure 5 shows the comparison of OMNET++ simulation results for the network connectivity of the schemes proposed in [10, 14, 15, 19].

Table 3: Memory cost.

Scheme	80% network connectivity			90% network connectivity		
	No. of keys	Key length (bits)	Memory used (bits)	No. of keys	Key length (bits)	Memory used (bits)
Proposed	2	160	320	2	160	320
Sarmad	20	160	3200	50	160	8000
Zhang	135	160	21600	150	160	24000
X. Du	225	160	36000	300	160	48000
E.G.	400	160	64000	500	160	80000

5.2. Memory Cost. This section presents a comparison of the memory cost of the proposed scheme with some well-known existing key management schemes for HSNs.

In an ECC-based key management scheme [2], the total memory overhead is $(n_{MN} + 3) * n_{FN} + 2n_{MN}$, where n_{MN} and n_{FN} are the numbers of MNs and FNs, respectively (as derived in [26]). Instead, in the solution presented by Yang et al. [20], each FN is preloaded with a pair of public/private keys and $n_{FN} - 1$ distinct pairwise keys, while no key is preloaded in the MNs. The memory overhead of this scheme is $(2 + n_{FN} - 1) * n_{FN}$. According to the basic scheme [10], each node is loaded with q keys before deployment, thus resulting in a total memory overhead of $q * (n_{FN} + n_{MN})$.

In the scheme proposed in this paper, each MN is loaded with only 3 keys (i.e., SK, K_{plc}, and K_{auth}) and each FN is loaded with 6 keys (i.e., the BS public key, its own public/private key pair, SKG, CNDK, and K_{prt}). The resulting memory overhead is $6n_{FN} + 3n_{MN}$.

To analyze and compare the proposed scheme with the existing schemes [2, 10, 14, 15, 20], it is assumed that each FN is able to make a maximum of d connections with its neighboring MNs. According to [10, 14, 15], if a node has N_c neighbors and that node has to establish secure links with only d neighbors, then the required key sharing probability should be

$$\Pr = \frac{d}{N_c}. \tag{6}$$

For example, the single key sharing probability required to make 30 connections with the neighboring MNs out of 38 neighbors is approximately 0.80. From Table 3, each node in [10] should carry 400 keys and each FN in [14] should carry 700 keys while each MN should carry 228 keys while in [15] each FN should carry 250 keys and each MN should carry 30 keys. In our scheme, each FN should be loaded with only 6 secret keys.

Table 3 summarizes the performance offered by different solutions in terms of the total number of the keys deployed for different sizes of the WSN. The results show that the proposed scheme requires definitely fewer keys compared to other approaches, especially in case dense networks are taken into consideration. For less dense networks, the proposed scheme and Yang's scheme require almost the same number of keys.

Since Zhang et al. [30] have presented the perturbation-based key establishment scheme for secure communication, its computational complexity and memory requirement increase with the increase in polynomial degree "t." Since each node is assigned a number of polynomials based on the required size of secret key and unaffected bits in perturbated polynomials, it increases the storage requirements exponentially. Also the receiver needs to find three different keys from the assigned perturbated polynomials in order to pick the correct one for the secure communication. However this approach increases the computational cost when key is combination of more than one perturbated polynomial. In our proposed scheme, we only need to assign few keys (shown in Table 3) for secret key establishment and secure communication.

Yu [34] proposed noninteractive pairwise key establishment schemes called constrained random perturbation-based pairwise key establishment (CARPY) and CARPY+ for secure communication. Although this scheme generates less commutation and communication overhead compared to [30], still it has large computation overhead compared to the proposed scheme. It is the improved version of Blom's scheme based on the concept of matrix perturbation. Only $l - r$ bits are used to construct secret key from different matrix polynomials where l is the minimum number of bits to represent an element in field F_q and r is the least bit perturbed. If L is the length of secret key, then $[L/(l - r)]$ rounds are required to generate a key, which reflect the computation cost of this approach. This approach is also memory expensive one to store large matrix for key establishment compared to the proposed approach.

5.3. Network Resilience to Node Compromised Attacks. This section shows the effect of node compromised attacks on data communication capabilities. In the proposed scheme, FNs and MNs are provided with different security measures dealing with such attacks. Since FNs act as both CHs and data sinks for MNs, they are provided with tamper-resistant hardware to protect their security material [35]. Once the FN is captured, all its security keys are replaced by a reference "compromised key" which neither allow the node to authenticate itself to the BS nor accept any joining MN. On the contrary, we assume that for cost reasons the MNs are not provided with the tamper-resistant hardware.

Node compromised attacks have a significant impact on the security offered by the communication links operating within the network in case of balanced and unbalanced key predistribution schemes for homogeneous and heterogeneous sensor networks due to the large number of shared keys with other nodes in the network. The fraction of

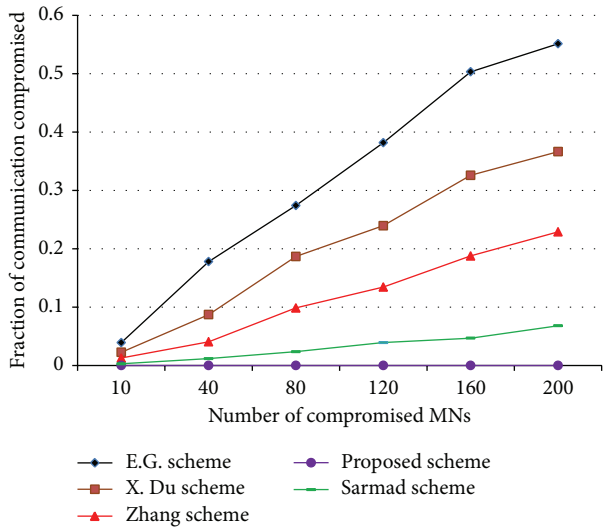

Figure 6: Fraction of communication compromised by capturing "n" mobile nodes (MNs).

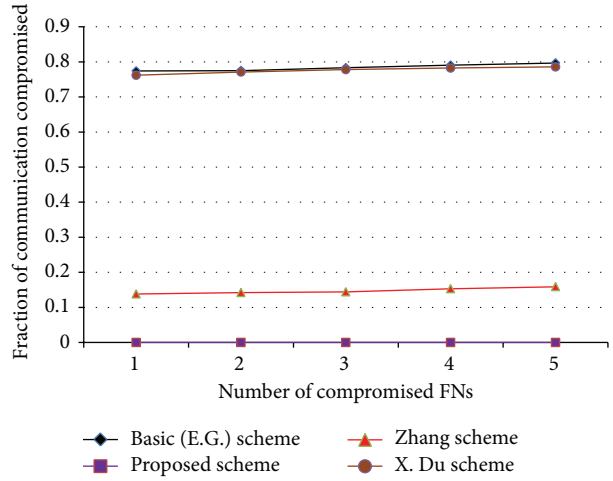

Figure 7: Fraction of communication compromised by capturing "n" fixed nodes (FNs).

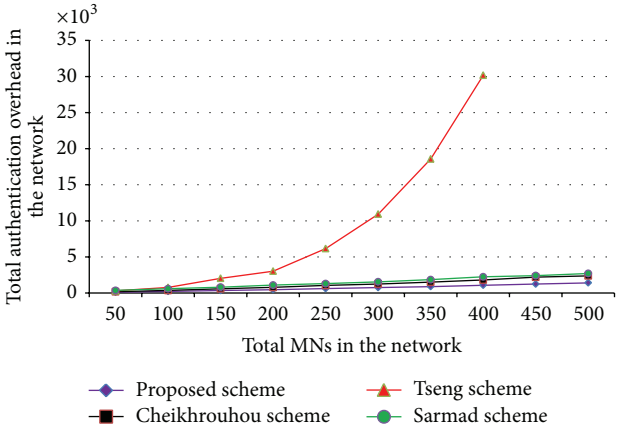

Figure 8: Authentication overhead.

communications compromised by compromising "n" MNs in shared key predistribution schemes is given by

$$\Pr\left[\text{Compromised}\right] = 1 - \left(1 - \frac{K}{P}\right)^{n}, \tag{7}$$

where K is the number of keys assigned to each MN from a pool of P keys. In case of compromised FNs, K is replaced by S in (7). Figure 6 shows the OMNET++ simulation results about how many communications links a compromised MN can create with uncompromised MNs without involving the CH/FN. More specifically, the figure compares the proposed scheme with the schemes proposed in [10, 14, 15, 19] with Pr[Conn] = 0.8. The proposed scheme performs better because (i) a MN cannot establish directly a communication link with the other MNs of the network and (ii) all the FNs use the algorithm proposed in [29] to detect the compromised MNs.

Since the FNs act as trusted servers to the MNs, their compromise can severely affect the network security. Figure 7 shows a comparison among the OMNET++ simulation results for the FNs compromise in the proposed scheme and in other reference solutions, that is, [10, 14, 19]. It is clear from Figure 7 that FN compromise results in almost the same number of compromised links when using [10, 14]. Although [19] proposed a balanced key distribution for the HSNs like [10] for homogeneous sensor networks, it performs better than [10, 14] because it divides the key pool P into a number of groups equal to the number of clusters, thus increasing not only network connectivity but also network resilience against both FN and MN capture attacks.

5.4. Communication Overhead. In this section, the communication overhead is evaluated also analyzing the different contributions from the authentication and key establishment phases. The simulation scenario is kept constant, with 16 FNs and 500 MNs.

5.4.1. Authentication Overhead. Concerning the authentication overhead, the total number of packets exchanged during the authentication phase is considered. The authentication phase of the proposed solution is compared with some of the existing approaches including [21, 22, 26]. OMNET++ simulation results show that the proposed scheme produces less authentication overhead than the existing schemes, as shown in Figure 8.

5.4.2. Key Establishment Overhead. As far as key establishment overhead is concerned, the proposed solution is compared with basic homogeneous [10] and heterogeneous [14, 26] schemes. The results show a significant reduction of the communication overhead. A 99% network connectivity probability for [10, 14] was taken into account computing the number of keys required in each FN and MN (using the results of (3) and (4)). The obtained results are shown in Figure 9. There is only a slight difference in terms of communication overhead between the homogeneous and heterogeneous approaches, but there is a big difference in terms of memory cost (Table 3).

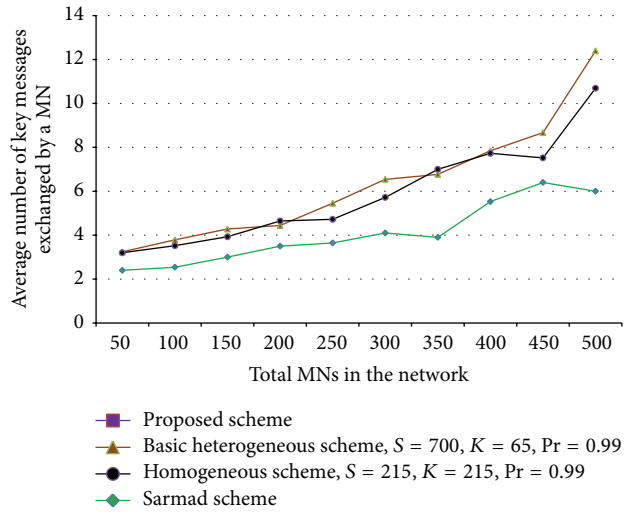

Figure 9: Key establishment overhead.

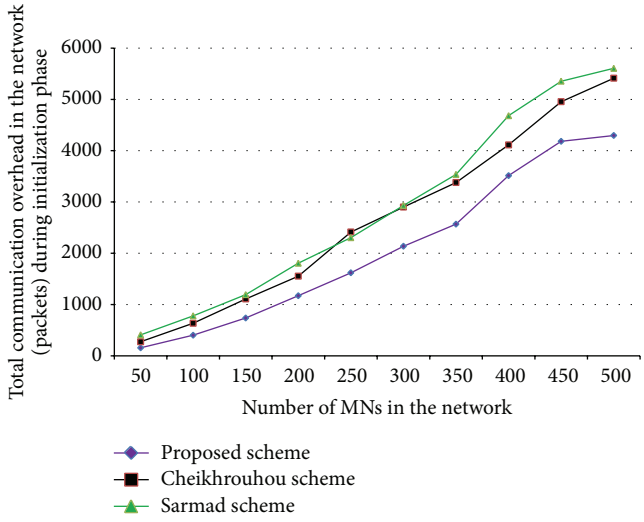

Figure 10: Total communication overhead.

5.4.3. Total Initialization Phase Overhead. This section presents the OMNET++ simulation results for the total communication overhead generated during the first authentication and key establishment phase. The results of the proposed scheme have been compared with the ones related to [21, 26], since both solutions are based on the mutual authentication and key establishment phases. Figure 10 represents the resulting communication overhead by varying the size of the network.

5.5. Energy Consumption. This section describes the average energy consumption of each node during the authentication and initialization phases of the network (again using the OMNET++ simulator). The proposed solution requires only 2 messages for the authentications as shown in Figure 8, compared with [22] which requires 4 messages and with [21, 26] which require 3 messages for authentication.

Figure 10 also shows the effectiveness of combining the authentication and key establishment phases to reduce the total overhead during the initialization phase. Such optimization results in power savings at each node and in an overall increase of the network lifetime. Figure 11 represents the OMNET++ results for the average energy consumption of each node during the initialization phase (authentication and key establishment) in the proposed scheme, as compared with [21, 26]. The results show that the proposed solution by combining the authentication and key establishment messages actually reduces the energy consumption with respect to [21, 26] where separate messages are exchanged for key establishment between the nodes after their successful authentication.

5.6. Overhead Generated during Handover. We have also analyzed the proposed scheme using OMNET++ simulator to check overhead generated during the handover phase. Figure 12 shows the comparison of the average number of messages exchanged to select the new CH and leave the old CH while Figure 13 shows the average overhead generated during this phase. We analyzed the network having 10 nodes and 50 nodes and keeping the node's speed of 1 mps, 5 mps, and 10 mps.

6. Conclusion

In this paper, a key management scheme is proposed for cluster-based heterogeneous sensor networks. In comparison with existing approaches, the proposed solution provides better network connectivity, reduces memory overhead, increases network resilience against node capture attacks, and requires minimum communication overhead during the authentication and key establishment phases. Hence it saves battery energy and increases the network lifetime. Also the security of the proposed scheme has been analyzed using the AVISPA tool against well-known attacks. In this paper,

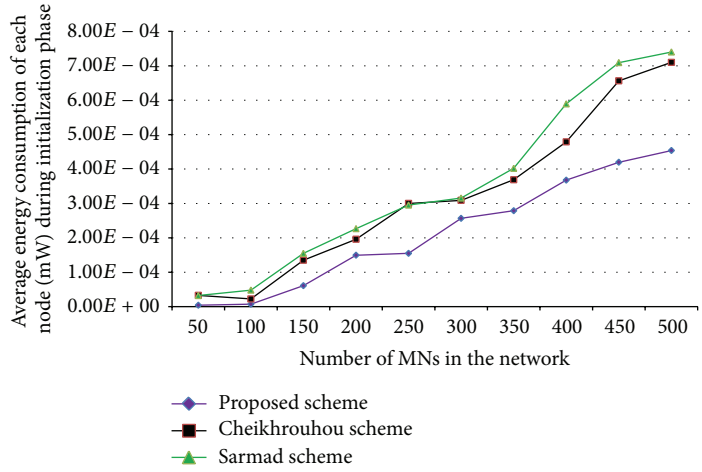

Figure 11: Average energy consumption.

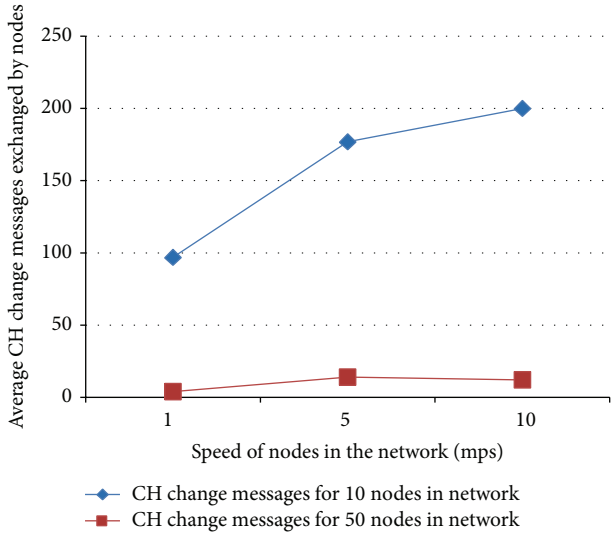

Figure 12: Average CH changed messgaes exchanged.

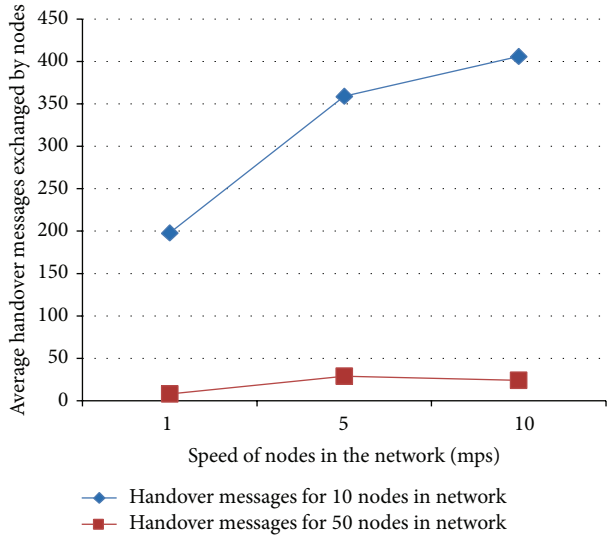

Figure 13: Average overhead generated during handover.

only intranetwork movements of the mobile nodes were considered. Future work will analyze internetworks mobility scenarios.

Conflict of Interests

The authors declare that there is no conflict of interests.

Acknowledgment

This paper has been partially supported by the European FP7 Project BUTLER, under Contract no. 287901.

References

[1] F. Liu, J. Rivera, and X. Cheng, "Location-aware key establishment in wireless sensor networks," in *Proceedings of the International Wireless Communications and Mobile Computing Conference (IWCMC '06)*, pp. 21–26, July 2006.

[2] X. Du, Y. Xiao, S. Ci, M. Guizani, and H.-H. Chen, "A routing-driven key management scheme for heterogeneous sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp. 3407–3412, June 2007.

[3] E. J. Duarte-Melo and M. Liu, "Data-gathering wireless sensor networks: organization and capacity," *Computer Networks*, vol. 43, no. 4, pp. 519–537, 2003.

[4] K. Xu, X. Hong, and M. Gerla, "An ad hoc network with mobile backbones," in *Proceedings of the IEEE International Conference on Communications (ICC '02)*, vol. 5, pp. 3138–3143, New York, NY, USA, May 2002.

[5] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[6] L. Girod, T. Stathopoulos, N. Ramanathan et al., "A system for simulation, emulation, and deployment of heterogeneous

sensor networks," in *Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems*, ACM Press, 2004.

[7] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, "Exploitingheterogeneity in sensor networks," in *Proceedings of the 24th Annual Joint Conference of theIEEE Computer and Communications Societies (INFOCOM '05)*, vol. 2, pp. 878–890, Proceedings IEEE, March 2005.

[8] E. J. Duarte-Melo and M. Liu, "Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '02)*, vol. 1, pp. 21–25, November 2002.

[9] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[10] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November 2002.

[11] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy (SP '03)*, pp. 197–213, May 2003.

[12] M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," Tech. Rep. MIT/LCS/TR-212, Laboratory for Computer Science, MIT, 1979.

[13] J. Zhang, Y. Sun, and L. Liu, "NPKPS: a novel pairwise key pre-distribution scheme for wireless sensor networks," in *Proceedings of the IET Conference on Wireless, Mobile and Sensor Networks (CCWMSN '07)*, pp. 446–449, December 2007.

[14] X. Du, Y. Xiao, M. Guizani, and H.-H. Chen, "An effective key management scheme for heterogeneous sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 24–34, 2007.

[15] S. U. Khan, L. Lavagno, and C. Pastrone, "A key management scheme supporting node mobility in heterogeneous sensor networks," in *Proceedings of the 6th International Conference on Emerging Technologies (ICET '10)*, pp. 364–369, October 2010.

[16] S. A. Çamtepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 2, pp. 346–358, 2007.

[17] D. Sánchez and H. Baldus, "A deterministic pairwise key predistribution scheme for mobile sensor networks," in *Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks (SecureComm '05)*, pp. 277–288, September 2005.

[18] J. Maerien, S. Michiels, C. Huygens, and W. Joosen, "MASY: management of Secret keYs for federated mobile wireless sensor networks," in *Proceedings of the 6th Annual IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '10)*, pp. 121–128, Niagara Falls, NY, USA, October 2010.

[19] J. Zhang and L. Zhang, "A key management scheme for heterogeneous wireless sensor networks based on group-oriented cryptography," in *Proceedings of the International Conference on Internet Technology and Applications*, pp. 1–5, Wuhan, China, 2010.

[20] Q. Yang, Q. Li, and S. Li, "An efficient key management scheme for heterogeneous sensor networks," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '08)*, pp. 1–4, Dalian, China, October 2008.

[21] O. Cheikhrouhou, A. Koubâa, M. Boujelben, and M. Abid, "A lightweight user authentication scheme for wireless sensor networks," in *Proceedings of the ACS/IEEE International Conference on Computer Systems and Applications (AICCSA '10)*, pp. 1–7, May 2010.

[22] H. R. Tseng, R. H. Jan, and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks," in *Proceedings of the 50th Annual IEEE Global Telecommunications Conference (GLOBECOM '07)*, pp. 986–990, November 2007.

[23] T. K. Kyeong and R. S. Ramakrishna, "A level-based key management for both in-network processing and mobility in WSNs," in *Proceedings of the IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems (MASS '07)*, pp. 1–8, Pisa . Italy, October 2007.

[24] I.-H. Chuang, W.-T. Su, C.-Y. Wu, J.-P. Hsu, and Y.-H. Kuo, "Two-layered dynamic key management in mobile and long-lived cluster-based wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 4148–4153, March 2007.

[25] C. Blundo, A. de Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Advances in Cryptology—CRYPTO '92*, vol. 740 of *Lecture Notes in Computer Science*, pp. 471–486, Springer, Berlin, Germany, 1993.

[26] S. U. Khan, L. Lavagno, C. Pastrone, and M. Spirito, "An effective key management scheme for mobile heterogeneous sensor networks," in *Proceedings of the International Conference on Information Society (i-Society '11)*, pp. 98–103, London, UK, June 2011.

[27] S. Hussain, F. Kausar, and A. Masood, "An efficient key distribution scheme for heterogeneous sensor networks," in *Proceedings of the International Wireless Communications and Mobile Computing Conference (IWCMC '07)*, pp. 388–392, August 2007.

[28] A. S. Poornima and B. B. Amberker, "Tree-based key management scheme for heterogeneous sensor networks," in *Proceedings of the 16th International Conference on Networks (ICON '08)*, pp. 1–6, New Delhi, India, December 2008.

[29] X. Jin, P. Putthapipat, D. Pan, N. Pissinou, and S. K. Makki, "Unpredictable software-based attestation solution for node compromise detection in mobile WSN," in *Proceedings of the IEEE Globecom Workshops (GC Wkshps '10)*, pp. 2059–2064, Miami, Fla, USA, December 2010.

[30] W. Zhang, M. Tran, S. Zhu, and G. Cao, "A random perturbation-based scheme for pairwise key establishment in sensor networks," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07)*, September 2007.

[31] G.-A. Kamendje, "A tamper resistant CMOS crypto-key generation unit," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '02)*, vol. 2, pp. II-352–II-355, Phoenix-Scottsdale, Ariz, USA, May 2002.

[32] http://www.avispa-project.org/.

[33] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil attack in sensor networks: analysis & defenses," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 259–268, IEEE, April 2004.

[34] C.-M. Yu, C.-S. Lu, and S.-Y. Kuo, "Noninteractive pairwise key establishment for sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, pp. 556–569, 2010.

[35] R. Anderson and M. Kuhn, "Tamper resistance: a cautionary note," in *Proceedings of the 2nd USENIX Workshop on Electronic Commerce (WOEC '96)*, vol. 2, p. 1, USENIX Association, Berkeley, Calif, USA, 1996.